

# 벡터기반의 캐리커처 자동생성에 관한 연구

박 연 출<sup>†</sup> · 오 해 석<sup>††</sup>

## 요 약

본 논문에서는 얼굴의 특징정보를 추출하여 사람의 얼굴과 유사한 캐리커처를 자동으로 생성해 주는 시스템을 제안한다. 본 논문에서 제안하는 시스템은 벡터를 기반으로 하기 때문에 사이즈에 제한 없이 자유로운 변형이 가능할 뿐만 아니라 2D 캐릭터에 자유로운 표정을 적용하는 데에도 쉽게 적용이 가능하다. 또, 벡터의 특징으로 인해 모바일 상에서도 적은 용량으로 이용가능하다.

## A Study on Vector-based Automatic Caricature Generation

Yeon-Chool Park<sup>†</sup> · Hae-Seok Oh<sup>††</sup>

### ABSTRACT

This paper proposes the system to generate caricature (character's face) resembling human face using extracted facial features automatically. Since this system is vector-based, the generated character's face has no size limit and constraint. so it is available to transform the shape freely and to apply various facial expressions to 2D face. Moreover, owing to the vector file's advantage, it can be used in mobile environment as small file size.

**키워드 :** 얼굴 검출(Face Detection), 얼굴 특징추출(Facial Feature Extraction), Facial Caricaturing

### 1. 서 론

최근 인터넷이 보편화 되고, 멀티미디어 콘텐츠 사업의 비중이 증대됨에 따라 가상공간에서 나를 대신하는 아바타(Avatar) 및 캐릭터 사업이 점차 큰 부가가치 사업으로 여겨지고 있다. 최근에는 캐릭터에 감정을 부여하여 감정 에이전트(Emotional Agent)로도 이용하고, 온라인상에서 제작된 캐릭터를 수천가지가 넘는 기존의 오프라인 캐릭터 상품으로까지 제작, 그 영역을 확장하여 그 활용 분야도 다양해지고 있는 추세이다.

본 논문에서 제안하는 캐리커처(또는 캐릭터 얼굴; 본 논문에서는 혼용해서 사용함) 자동생성 시스템은 연예인이나 본인의 사진으로부터 사람의 얼굴과 유사한 캐리커처(반실사형 캐릭터)를 자동으로 생성한다. 여기서 생성된 캐리커처(반실사형 캐릭터)를 기존 만화형 캐릭터 대신 이용할 경우 마치 그 캐릭터가 실제 본인과 같은 일체감을 얻을 수 있고, 반면 사진과 달리 실제 얼굴이 공개되지 않으므로 익명성도 보장하여 인터넷에 적합한 대체 ID로 적절히 이용할 수 있다는 장점을 갖는다.

그러나 이러한 장점에도 불구하고 캐릭터 제작 과정이 전문 디자이너에 의해 수작업으로 이루어지고 있어 많은 인력과 시간 낭비를 초래하고 있다.

본 논문은 이렇게 수작업으로 이루어지고 있는 캐릭터 생성 작업을 입력된 얼굴 이미지로부터 특징을 추출하고, 얼굴 합성기법을 이용하여 자동화 해주는 시스템을 제안한다.

본 논문은 크게 입력된 정면의 사진에서 특징정보를 추출하는 과정과 데이터베이스에서 유사한 이목구비 이미지를 추출하여 매핑시키는 과정으로 나누어진다. 구체적으로 설명하면 다음과 같다.

첫째, 정면의 입력된 인물 사진에서 얼굴 영역을 추출한다. 둘째, 얼굴영역에서 수직, 수평 히스토그램을 이용하여 이목구비(눈, 코, 입, 눈썹) 등을 추출한다. 셋째, 추출된 이목구비에서 위치, 크기, 모양 등 다양한 특징정보를 추출한다. 넷째, 특징정보를 이용하여 데이터베이스에 저장되어 있는 이목구비 이미지와 추출된 이미지의 유사도를 측정한다. 다섯째, 가장 유사한 이목구비 이미지를 추출하여 각각 눈, 코, 입, 눈썹, 얼굴형 등을 매핑시킨다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 캐리커처 생성 시스템을 분석하고, 기존 방법의 문제점 분석 및 개선 방안 살펴보면, 3장에서는 캐리커처 생성 시스템의 구조 및 모듈별 기능 및 연구 성과에 대해 설명하고, 4장에서

※ 본 연구는 숭실대학교 교내 연구비 지원으로 이루어졌음.

† 준 회 원 : 숭실대학교 대학원 컴퓨터학과

†† 종 신 회 원 : 숭실대학교 컴퓨터학과 교수

논문접수 : 2003년 5월 20일, 심사완료 : 2003년 8월 9일

는 이 논문의 실험결과 및 분석에 대해 논하고 5장에서는 결론을 맺는다.

## 2. 캐리커처 생성 시스템

### 2.1 후쿠이(Fukui) 대학의 캐리커처 생성 시스템

후쿠이 대학(Fukui University)의 캐리커처 생성 시스템은 사람의 생각의 애매함과 복잡함을 단순화시켜 처리하는 퍼지 셋을 이용하여 캐리커처를 생성하는 시스템이다[1, 5-7].

표준 얼굴 이미지의 21개의 특징(눈의 크기, 눈썹두께, 코의 위치, 높이, 얼굴형태, 머리 길이 등)에 파라미터를 이용하여 입력된 이미지의 특징 정보를 추출한다. 이렇게 추출된 각각의 크기를 49개의 특징 단어(big, small, round, thin, tall, short 등)를 이용하여 단어들의 분석에 따라 선택하여 얼굴에 배치시킴으로써 캐리커처를 생성한다.

### 2.2 추쿄(Chukyo) 대학의 PICASSO System

추쿄 대학(Chukyo University)의 PICASSO System은 생성된 캐리커처의 감정 표현을 목적으로 개발된 시스템이다. 이 시스템의 캐리커처 생성 방법은 표준 얼굴 이미지의 윤곽선을 455개의 포인트로 표현하고 입력된 이미지에 455개의 포인트를 이용하여 각각의 포인트를 비교하여 그 포인트를 확대와 축소로 조절하여 포인트를 선으로 이어 윤곽선을 그려줌으로써 캐리커처를 생성한다[2, 7].

PICASSO SYSTEM은 표준 얼굴 이미지의 특징(이목구비)과 얼굴 윤곽선을 455개의 포인트에 의해 표현하고, 입력된 얼굴 이미지 또한 포인트로 표현한다. 그리고 표준 얼굴 이미지와 입력 이미지와의 각 포인트를 비교한다. 비교된 포인트를 확대와 축소로 조절하여 입력된 이미지의 윤곽선을 포인트로 표현한다. 이렇게 처리된 455개의 포인트를 서로 이어 윤곽선으로 나타내어 캐리커처를 생성한다.

### 2.3 기존 시스템의 문제점 및 개선 방안

기존 시스템의 문제점은 다음과 같다. 첫째 다양하지 않은 특징정보를 이용하여 생성하므로 얼굴의 다양성이 반영되지 않았다는 것이다. 둘째 표준 얼굴 이미지를 기반으로 변형작업을 통하여 캐리커처를 생성하므로 입력된 이미지의 윤곽선을 그대로 그려주는 것에 불과하여 이미지 변형 효과를 주었다는 것이다. 이러한 문제점에 대해서 자세히 살펴해보도록 한다.

Facial Caricature Drawing System은 사람의 다양한 생김새를 단순한 크기를 이용하여 생성한다. 그래서 사람이 보기에는 확연히 다른 얼굴도 같은 특징 언어(feature terms)를 이용하기 때문에 같은 캐릭터로 여겨질 수 있다. 즉, 변화율이 그다지 크지 않다.

PICASSO System은 표준 얼굴 이미지를 이용하여 455

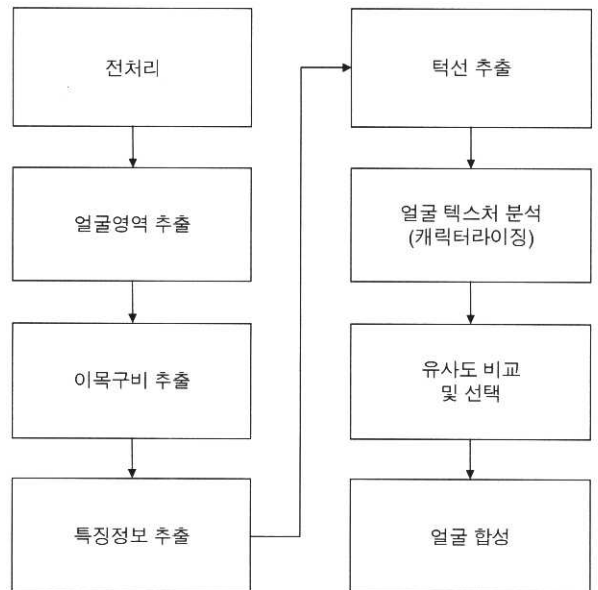
개의 포인트를 갖은 윤곽선에 대한 입력 이미지에 455개의 포인트를 확대/축소로 생성하는 캐리커처이므로 다양한 캐리커처를 생성하지만 입력된 이미지의 윤곽선을 이용하여 캐릭터를 그려주므로 입력된 사진의 윤곽선만을 이용한 변형에 불과하다.

하지만, 엄밀히 따지자면 위 시스템들은 닳은 얼굴의 구성에 초점을 맞춘 것이 아니라 얼굴의 풍자 및 희화화에 초점을 맞추고 연구 진행되어져 온 것이므로 본 논문이 제안하는 시스템과는 다소 거리가 있다.

## 3. 캐리커처 자동생성 시스템

### 3.1 시스템 구조

(그림 1)은 캐리커처 생성 시스템의 구조와 데이터 흐름을 함께 표현한 것이다.



(그림 1) 시스템 구성도

입력된 이미지에 캐리커처 생성 과정을 간략하게 설명하면 다음과 같다.

- ① 영역 추출
  - 얼굴 영역 추출 : 명도와 색상 정보 이용
  - 부분 영역 추출 : 히스토그램 이용
- ② 동일 class 검색
  - 얼굴 특징(face features) 정보를 이용
  - D/B에 분류된 정보를 이용
- ③ 유사도 계산
  - 유사한 이목구비 이미지 추출
- ④ 추출된 캐릭터 이미지 매핑
  - 추출된 특징정보 위치에 이목구비 매핑

본 시스템의 입력 이미지는 정면을 보고 있는 상반신 사진이다. 입력된 이미지에서 얼굴 영역을 추출하고, 이어서 이목구비(눈, 코, 입, 눈썹)의 부분영역을 추출하고, 추출된 이목구비로부터 특징정보를 추출하고, 미리 구축해 놓은 이목구비 특징정보 데이터베이스를 이용하여 각각의 영역에서 가장 유사도가 높은 이미지를 검색해 낸다. 본 논문에서 이용하는 이목구비 데이터베이스는 전문 디자이너에 의해 그려진 이목구비를 각각의 특징에 맞게 특징정보와 함께 분류해 놓은 것이다. 추출된 각 이목구비를 얻어진 특징정보에 의해 매핑시켜, 사용자에게 보여줄 수 있도록 설계되었다.

### 3.2 전처리

본 시스템에서 행하고 있는 전처리로는 가우시안 스무딩이 전부이다. 왜냐하면 그 외에 필요한 우선 처리 과정은 모두 각 스텝에서 필요시마다 별도로 행해지고 있기 때문이다. 본 논문에서 전체 이미지에 대해 필요한 전처리가 있다면 잡음제거가 전부이다. 따라서, 잡음제거를 위해 본 논문에서는 대표적인 가우시안 스무딩을 사용하였다.

### 3.3 2단계 얼굴영역 추출

얼굴 영역을 추출하기 위해서는 사람의 얼굴이 갖는 컬러의 분포 형태를 가정하는 것이 우선 필요하다. 이에 본 논문에서는 약 200장의 사진으로부터 HSI 컬러 스페이스의 벡터값( $h, s$ )을 얻어내어 이를 수치화 하는 작업을 수행하였다.

지금까지는 컬러 차트를 이용하여 컬러값을 얻어내는 방식을 사용하였으나 이 방식은 시간과 계산량이 많이 요구되는 단점을 가지고 있다. 이에 본 논문은 이전의 많은 논문이 사람의 얼굴이 갖는 컬러가 정규분포를 띤다고 가정하였으므로 굳이 컬러 차트를 사용할 필요가 없다는데 착안하여 평균과 분산(표준편차)만으로 사람의 얼굴의 분포를 수치화하여 사용하였다. 이는 기존의 논문이 얼굴의 영역을 찾아낼 때 유사한 얼굴 영역을 찾아내는 과정에만 컬러 차트를 사용하였는데 이는 계산량이 증가하여 결국에는 전체적인 분석시간의 증가만 초래하였다. 하지만, 본 논문이 제안하는 방식은 1단계 추출에서는 컬러 분포를 정규분포라 가정하고, 평균과 표준편차를 이용하여 신뢰구간내의 유사 얼굴 영역을 추출하고, 그 영역내에서 일단 눈동자를 찾은 후, 단일 사진내에서 추출된 유사 얼굴 영역내에서 다시 한번 샘플링을 하여 그것을 차트화 하고 이를 통해 정확한 얼굴 영역을 얻어내는 2단계 방식을 사용하였다.

이와 같은 방법을 사용하는 이유는 사람의 얼굴이 갖는 얼굴 컬러의 분포는 어느 정도 유사하지만 실제로 각 사진마다 갖는 얼굴의 분포는 사진별로 조금씩 차이를 보이기 때문이다. 이 차이는 각 사진마다 조명, 화장, 카메라의 종

류, 찍는 사람의 능력에 따라 다르게 나타난다. 즉, 각 사진이 갖는 특성을 그대로 적용하여 얼굴을 추출하기 위해서는 그 사진내에서 직접 정보를 얻어내어야 한다는 결론을 얻을 수 있다. 이에 본 논문은 1차적으로 통계적 모멘트 값을 이용한 얼굴 영역의 검출 후, 2차적으로 다시 사진내에서의 얼굴의 컬러 분포의 모멘트 값을 얻어내어 얼굴의 영역 및 윤곽을 추출해 내는 2단계 컬러 샘플링 방식을 사용하였다.

본 논문에서 HSI Space를 사용한 이유는 알려진 바와 같이 이 방식이 사람 시각의 컬러 인지 방식과 가장 흡사하기 때문이다.

얼굴 컬러의 추출은 HSI 값 중에서 HS 값만을 벡터 형태로 취하였다. N이 샘플의 총 개수라고 할 때, 샘플 하나의 값은  $(h_i, s_i), i=1, \dots, N$  이다. 여기서 명암 값(Intensity)을 사용하지 않은 이유는 얼굴에서 명암값의 분포가 균일하지 못하고 얼굴 전체에 걸쳐 다양하게 퍼져있기 때문이다. 이는 조명이나 화장에 의해 영향을 많이 받기 때문이기도 하다. 따라서, 컬러값만을 사용하게 되면 조명이나 화장에 강한 장점을 갖는다.

얼굴색의 분포 모델을 얻기 위해 사용된 사진의 개수는 총 194장(남 : 97, 여 : 97)이며, 사진의 획득은 인터넷을 통해 이루어졌다. 모두 임의의 유저로부터 직접 제출받은 것들이다. 따라서, 랜덤하게 추출되어졌다고 볼 수 있다. 단, 사진의 크기(가로, 세로 100픽셀 이상)와 정면 사진이라는 전제가 있었다.

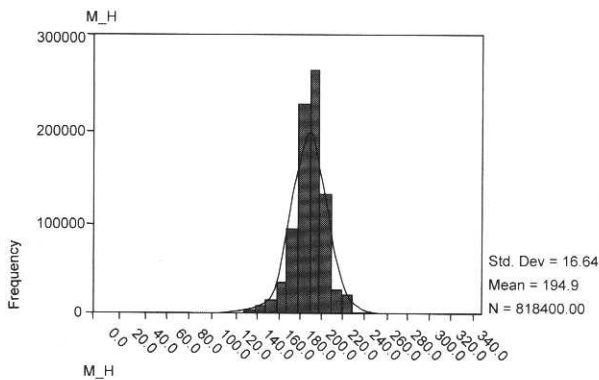
샘플링은 (그림 2)에서 보는 바와 같이 가능하면 얼굴 영역 전체에 걸쳐서 골고루 추출될 수 있도록 하기 위해, 사진에서 얼굴 부분 중 14부분을 선택하여 직사각형( $20 \times 30$ ) 형태로 추출하였다. 600픽셀, 총  $194 \times 14 \times 600 = 1,629,600$  픽셀. 이 값들을 이용해  $E(H), E(S), Var(H), Var(S)$ 를 구한 후, 이를 이용해 신뢰구간을 획득하였다.



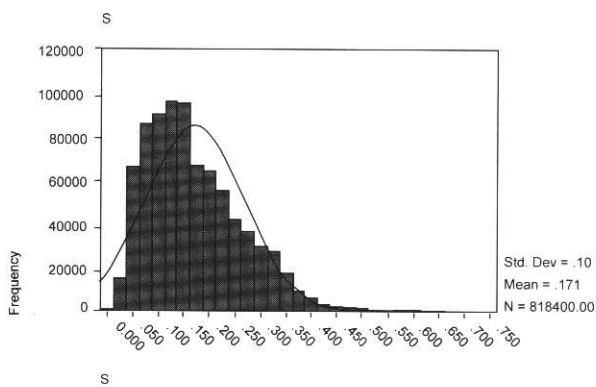
(그림 2) 살색 표본 추출

신뢰구간은  $\bar{x} - 3\sigma < \mu < \bar{x} + 3\sigma$ 이다. 여기서, alpha 값을 0.01로 했을 경우, 신뢰수준은 99%이고, 오차범위는 0.5%이다.

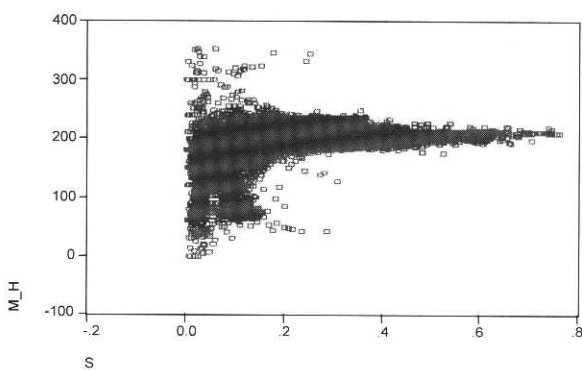
(그림 2)는 사진으로부터 살색을 샘플링 하는 과정을 표현한 것이다. 총 14개의 영역으로부터 샘플링이 이루어졌다. 이 14개의 영역의 선택은 얼굴 전체를 골고루 대표할 수 있고, 변화가 심하지 않은 곳이어야 한다는 기준에 의해 이루어졌다. 피부가 아닌 샘플이 추출될 가능성을 줄이기 위하여 자동 추출을 하지 않고 수작업으로 행하였다.



(그림 3) 변형된 Hue값의 히스토그램



(그림 4) Saturation값의 히스토그램



(그림 5) Hue와 Saturation 산포도

(그림 3)과 (그림 4)는 얻어진 데이터를 히스토그램 형태

로 표현한 것이다. 그림 안의 곡선은 정규분포를 표시한 것이다. 또, (그림 5)는 Hue와 Saturation값을 정규화하여 2차원 공간에 산포도로 표현한 것이다.

### 3.4 눈동자 추출

눈동자의 추출에 앞서 이목구비 영역의 추출이 선행되어야 하는데 명암의 수평, 수직 히스토그램 분석과 한국인의 평균 얼굴형 모델을 통하여 얻어질 수 있다[11]. 즉, 히스토그램을 이용하여 이목구비의 위치를 예측해 낸 후, 이 위치의 적합성의 판단을 위해 평균 얼굴형 모델의 수치가 사용되어졌다. 여기서 얻어진 눈의 부분영역이 눈동자의 추출 시 사용되어지는 입력 영상으로 사용되어진다.

눈동자의 추출은 얼굴인식 분야에서 중요한 파트중의 하나이다. 왜냐하면, 사진에서 얻어낼 수 있는 가장 신뢰할 만한 특징점일 뿐만 아니라 눈동자 사이의 거리가 여러장의 사진들을 정규화하는데 가장 기본이 되는 기준치가 되기 때문이고, 또한 사진 속 얼굴의 수평을 맞추는데도 사용되기 때문이다.

본 논문에서는 눈동자를 찾기 위해 템플릿 매칭을 사용하였다. 템플릿은 크기가 가변적인 정원을 사용하였다. 눈동자가 위치할 만한 후보 영역을 우선 찾아내어야 하는데 이는 명암값의 수직 수평 프로젝션을 통해 얻어낼 수 있었다. 이렇게 얻어진 좌측 눈 영역과 우측 눈 영역이 각각 템플릿 매칭이 이루어지는 영역으로 사용된다.

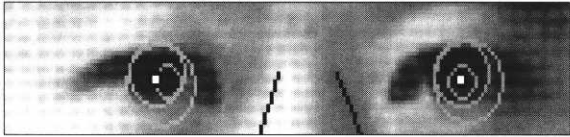
이렇게 얻어진 영역에 전처리 과정으로 눈동자에 반사된 빛을 제거하는 과정이 필요하다. 반사된 빛이 제거된 이미지는 3×3 마스크의 최소값을 이용하여 얻어낼 수 있다.

눈동자를 찾기 위해 사용된 템플릿의 수식은 다음과 같다.

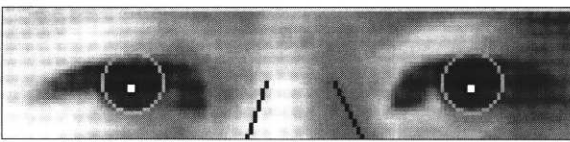
$$\begin{aligned}
 &x^2 + y^2 \leq r^2 \\
 &, \text{where } x = 1, 2, \dots, N1, \\
 &y = 1, 2, \dots, N2, \\
 &r = 15, 14, \dots, n
 \end{aligned}
 \tag{1}$$

여기서, x, y는 각각 이미지의 수평, 수직 인덱스이며, r은 반지름을 의미한다. 이렇게 얻어진 템플릿의 특징 벡터 (X, Y, R, S, V) 값을 이용하여 최적의 눈후보를 찾아내게 된다. 여기서, S는 반지름이 같은 템플릿 내에서의 최적의 후보를 찾는데 사용되는 모멘트 값이고, V는 반지름이 서로 다른 템플릿 중에서 최적의 후보를 찾기 위해 사용되는 모멘트 값이다. V값으로 본 논문에서는 2차 모멘트(분산)를 사용하였다. 분산을 사용한 이유는 눈은 하얀색 흰자와 검은색 눈동자로 구성되어지기 때문에 최적의 눈동자의 크기를 얻어냈다고 했을 때 명암의 변화율이 가장 낮기 때문이다. 또, 단순히 같은 반지름을 갖는 템플릿중에서 분산값이 가장 작은 템플릿을 선택하는 것이 아니라,  $r_{i-1}$ 의 분산과  $r_i$

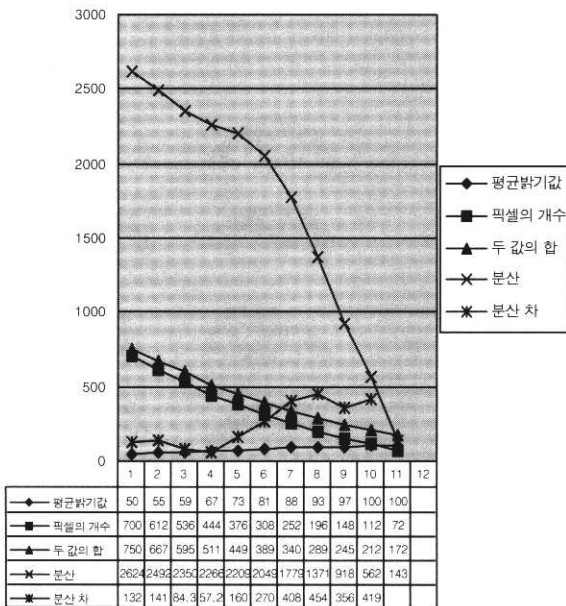
의 분산의 차가 가장 클 때의 원을 최적의 원으로 보았다. 이렇게 두 번의 비교 과정을 거침으로써 명암 이미지에서 검은 눈동자를 가장 최적으로 포함하는 눈동자의 후보를 찾아낼 수가 있다. (그림 6)은 템플릿의 반지름을 수렴시켜 가면서 S값을 비교, 값이 가장 큰 지점을 표시한 것이다.



(그림 6) 탐색 과정 (템플릿 사이즈의 수렴)



(그림 7) 추출된 눈동자의 위치와 크기



(그림 8) 분산과 분산차의 그래프

(그림 7)은 각 반지름별 최적 후보끼리 분산차를 구하여 분산차가 가장 큰 최종 최적해를 표시하고 있다. 눈동자의 위치와 크기를 동시에 얻어낼 수 있다는 것을 알 수 있다. (그림 8)은 템플릿의 각 반지름별 분산 및 반지름간의 분산차를 그래프 형태로 표현한 것이다. 보이는 바와 같이 그래프상에서 분산차가 최대가 되는 지점이 최적해로 선택되어진다.

이렇게 얻어진 눈의 좌표를 이용하여 이미지를 다시 회전시킨다. 회전을 시키는 이유는 얼굴의 수평을 맞추기 위해서이다. 눈동자가 얼굴 중에서 가장 찾기 쉬운 객체이고, 오인할 확률도 적기 때문에 눈동자가 항상 얼굴의 가장 기

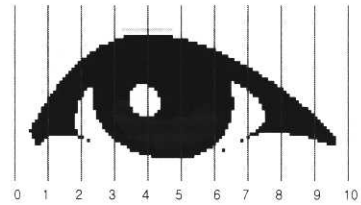
본적인 기준점이 된다.

### 3.5 이목구비의 특징정보 추출

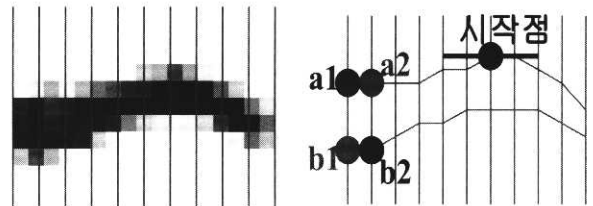
본 논문에서 우리는 추출한 이목구비를 이진화한 후 유사도 비교를 행한다. 이때, 이목구비 영상은 매우 작아서 약간의 잡음(noise)에도 민감하기 때문에 잡음제거가 우선적으로 필수적이다.

본 논문에서는 이목구비 영상 데이터의 에지를 흐리게 하지 않고 잡음을 제거할 수 있도록 가장 대표적으로 많이 쓰이는 미디언(Median) 필터를 전처리 과정에서 사용하였다. 전처리가 끝난 후에는 이진화 되어있는 이목구비 영상에 대한 특징정보를 추출한다.

(그림 10)은 이목구비 중 왼쪽 눈썹의 특징정보를 추출하기 위한 한 예이다. 우리는 그림처럼 왼쪽 눈썹을 가로로 10등분을 하고, 10등분된 선에 대한 각각의 시작점과 끝점, 그리고 한 점에서 다음 점까지((a1, a2)···, (b1, b2)···)의 기울기와 31개에 나타나는 눈썹 폭의 크기 등을 특징 벡터 형태로 표현, 눈썹의 특징 정보를 얻어낸다. 이때 사용되는 수치는 모두 가로폭으로 정규화된 값이다.



(그림 9) 눈의 특징정보 추출



(그림 10) 왼쪽 눈썹의 특징정보 추출

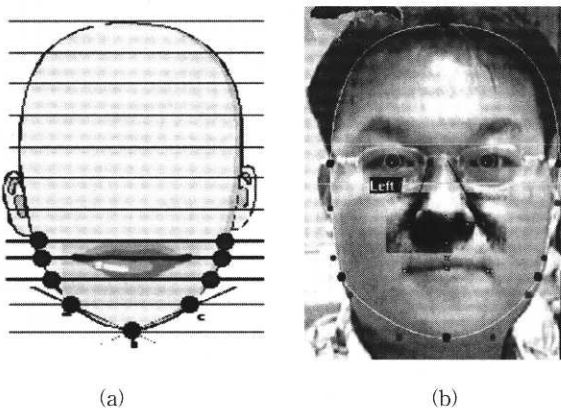
이와 같이 눈과 입술도 똑같은 형태로 특징추출이 행해진다. 입술 같은 경우는 이에 덧붙여 윗입술의 두께, 아랫입술의 두께 등도 특징정보로 추출되어진다.

### 3.6 턱선 특징정보 추출 및 벡터화

턱선의 특징정보는 우선 턱선의 형태를 벡터로 표현한 후, 이들 벡터 데이터를 표준 얼굴형과 비교함으로써 얻어낼 수 있다. 본 논문에서는 턱선의 형태를 벡터로 추출하기 위해 국소영역분할 방식을 이용하여 턱선의 특징점들을 추출해낸다. 이때 특징점들의 추출 오차를 줄이기 위하여 얼굴 내에서의 살색을 샘플링한 후에 그 데이터를 기초로

다시 세그먼테이션을 행하는 방식을 취하였다.

턱선의 특징점 추출은 음영차 (윤곽선)만을 이용하지 않고, 색상도 함께 이용하여 얼굴 경계 추출의 정확도를 높였다. 만약 음영만을 이용하였다면 살색과 유사한 헤어를 갖는 사진일 경우 헤어와 얼굴을 구분해 내지 못할 것이다. 물론 그 역의 경우도 발생 가능하다. 여기서 중요한 문제로 대두되는 것이 턱선의 특징점 후보들 중에서 최적해를 어떻게 찾아내느냐 하는 것이다. 즉, 음영과 컬러값들을 이용하여 세그먼테이션을 행할 경우 각각의 방법으로 얻어진 특징점 후보들이 항상 일치하지는 않는다. 따라서, 본 논문에서는 각 추출된 특징점 후보들과 표준 얼굴 모델의 특징점들과의 euclidean distance 값을 구하여 거리차가 가장 적은 특징점을 최적해로 사용하였다. 이때 사용된 표준 얼굴 모델의 특징점들은 두 눈동자 사이의 거리값에 의해 유추해낸 양 눈의 연장선이 만나는 얼굴의 끝점과 아래 턱점, 그리고, 윗입술의 상단, 입술의 중앙, 입술의 하단 점 등이다. 이렇게 얻어진 얼굴의 13개 특징점들을 Bezier Curve로 연결하여 얼굴형을 만들어낼 수 있었다 (그림 11). 만들어진 Bezier Curve는 쿼 이미지와 함께 벡터 파일(\*.EMF)로 저장된다.



(그림 11) (a) 턱의 특징점 (b) 턱선 및 이목구비별 특징정보 추출

3.7 이미지 분류(Classification)

일반적으로 이미지 분류란 검색 대상이 되는 이미지를 분류하여 검색되는 후보의 수를 줄이기 위하여, 물리적 대상이나 사건을 의미 있는 특징들의 추출을 기반으로 사전에 정의된 클래스 중 하나로 할당하는 것을 말한다. 본 논문에서 우리는 하나의 이미지로부터 추출된 다수의 특징정보에 따라 특정 클래스로 분류해 낸다. 이 분류의 기준은 위 3.5에서 설명한 것과 같이 각 이목구비의 특징정보를 이용한다. 그리고 본 논문에서 클래스화된 눈, 코, 입, 얼굴형, 눈썹들의 수는 <표 1>과 같이 각 부분별로 해당하는 수만큼 데이터베이스에 존재하고 있다. 입력된 얼굴 이미지는 이목구비의 특징정보를 이용하여 나누어진 다수의 클래스

에서 가장 유사도가 높은 것을 추출하여 이목구비를 매핑 시키게 된다. 다음은 얼굴형과 이목구비의 특징 정보별 클래스 분류를 설명한다.

<표 1> 벡터 이미지 개수

종 류	수 량
눈썹	각 200
눈	각 200
코	20
입	20

3.7.1 눈의 클래스 분류

눈의 클래스 분류는 눈썹과 마찬가지로 (그림 9)와 같이 10등분하여 특징정보를 추출하는데 10등분 중 몇 번째 위치에서 가장 높은 값을 갖는지에 따라 분류하고, 폭이 가장 큰 값이 얼마인지에 따라 큰 눈, 작은 눈을 분류한다. 이렇게 분류하면 <표 2>와 같이 나타나게 된다.

<표 2> 눈의 특징별 분류

크 기	Top Section	눈 의 종 류	
크 다	0~3		
	4~6		
	7~10		
작 다	0~3		
	4~6		
	7~10		

3.7.2 눈썹의 클래스 분류

<표 3> 눈썹 특징별 분류

	일자인 경우	좌측이 높은 경우	우측이 높은 경우	중심이 높은 경우
좌측이 두꺼운 경우				
일자인 경우				
우측이 두꺼운 경우				

눈썹의 경우, 다양한 특징정보를 이용하여 분류한다. (그림 10)의 눈썹의 특징정보 추출방식과 같이 10등분하여 최고점을 찾아내고 11개의 두께와 그 중 가장 두꺼운 부분, 그리고 올라가다 꺾어져 내려가는 점 등 매우 다양한 특징을 이용한다. 또한 어느 쪽이 두꺼운지 그리고 우측인지 좌측인지 등과 같이 다양한 특징을 추출하여 특징정보로 이용한다. <표 3>은 눈썹의 특징정보를 이용한 분류이다. 이와 같이 눈썹은 12개의 클래스로 분류되어 있다.

3.7.3 코의 클래스 분류

입력된 이미지에서 코는 사실 잘 추출되지 않는다. 아니, 추출되어도 얼굴색과 너무 같기 때문에 특징을 찾아내기가 가장 어렵다. 그래서 코는 콧구멍의 크기로 분류한다. 그러나 콧구멍도 잘 나타나지 않으므로 나타나지 않을 경우 일반적인 코를 추출한다. <표 4>는 콧구멍에 따라 분류한 코의 특징별 분류를 설명한 것이다.





<표 4> 코의 특징별 분류

일반적인 코	매부리 코	들창코
		

3.7.4 입의 클래스 분류

여성의 입 같은 경우 화장을 하기 때문에 사진에서 입을 추출하는 것이 그리 어렵지 않으나 남성은 그렇지 않으므로 얼굴색과 입술을 구분하기가 어렵다. 그래서 입도 코와 마찬가지로 나타나지 않을 경우 일반적인 입술을 추출하여 매핑시키게 된다. <표 5>는 추출된 입술의 두께에 따라 분류한 입술의 특징별 분류이다.

<표 5> 입술의 특징별 분류

얇은 입술	일반적인 입술	아랫입술 두꺼움	두꺼운 입술
			

3.8 검색 및 유사도 계산

일반적으로 이미지 검색은 유사 이미지를 검색하는 목적에 따라 색상 정보만을 적용할 때도 있고, 모양정보만을 적용할 때가 있다. 또는 두 특징정보의 적절한 배합으로 검색하고자 할 때가 있을 수 있다.

입력된 이미지에서 수직, 수평 히스토그램을 이용하여 추출되어진 이목구비 이미지의 색상은 다르고, 모양은 거의 같다. 그리하여 본 논문에서는 위와 같이 모양이나 색상을 이용한 특징정보를 이용할 수 없다. 그래서 본 논문에서는

특징정보 데이터와 이목구비 이미지와의 유사도를 계산하여 이미지 검색을 시도하였다.

본 논문에서 눈과 눈썹 이목구비 윤곽선의 유사도를 계산하기 위해 사용한 유사도 공식은 다음과 같다.

$$S = r \times \omega + (1 - |\alpha - \beta|) \tag{2}$$

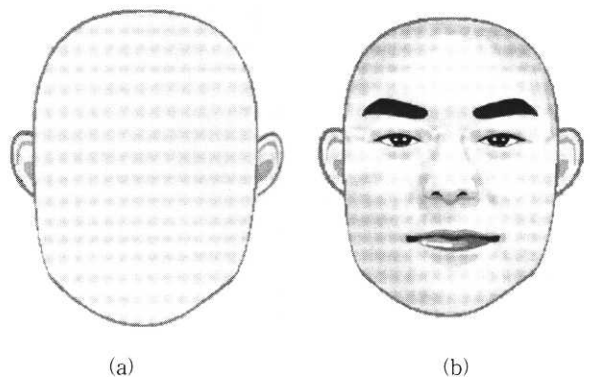
$$r = \frac{\sum(A_i - E_A)(B_i - E_B)}{\sqrt{\sum(A_i - E_A)^2 \sum(B_i - E_B)^2}} \tag{3}$$

여기서, 식 (3)은 상관계수이다. 이 식에서  $A_i$ 는 입력된 이미지에서 추출된 이목구비인 A의 구간별 높이를 의미하고  $B_i$ 는 데이터베이스에 있는 이미지인 B의 높이를 의미하며,  $E_A, E_B$ 은 A와 B의 평균 높이를 의미한다. 상관계수인 r의 성질은  $\pm 1$ 에 가까울수록 강한 상관관계를 갖는다. 하지만, 본 논문에서는 음의 상관관계는 의미가 없으므로 고려하지 않는다. 그러므로 r의 값은 0부터 1의 양의 실수 값을 갖는다. 또한 식 (2)의  $\alpha, \beta$ 는  $A_i, B_i$  각각의 가장 큰 가로와 세로의 비를 의미한다. 그리고  $\omega$ 는 가중치이다.

식 (2)의 값이 클수록 윤곽선 형태의 유사성이 높음을 의미한다.

3.9 이목구비 매핑

이목구비 매핑은 각각의 이목구비 이미지들을 얻어진 위치특징정보에 따라 적절히 배치시켜 캐리커처의 얼굴을 재구성하는 것을 의미한다. 입력된 얼굴 이미지에서 추출된 이목구비 데이터와 이목구비 데이터베이스에 저장된 각 벡터 이미지들 사이의 유사도를 계산하여 유사도가 가장 높은 각 이목구비를 검색하여 이미지를 매핑시키는 작업이다. 이 시스템에서 매핑시키는 위치는 입력된 데이터에서 추출한다. 즉 입력된 데이터의 이목구비를 추출할 때 이목구비의 위치를 찾아내고 그 위치정보를 이용하여 매핑시에 이용한다. 그러므로 각 개개인의 이목구비 위치 특징을 이용할 수 있다.



(그림 12) (a) 벡터로 만들어진 얼굴형 (b) 매핑된 얼굴형

(그림 12)에서 좌측의 그림은 (그림 11)에서 보여졌던 바

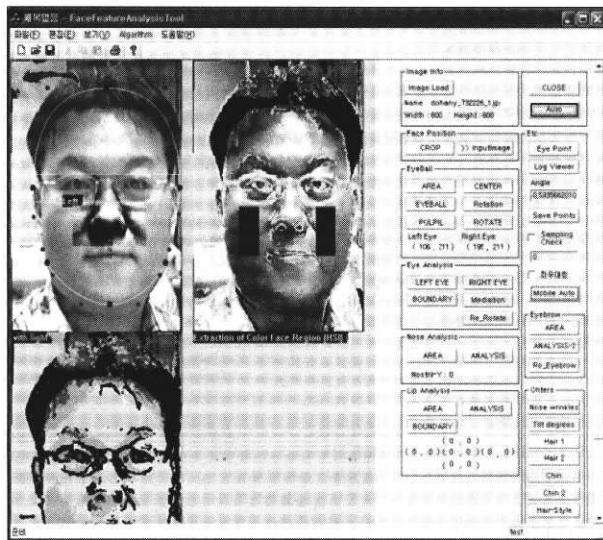
와 같이 턱선의 특징점들을 추출한 후 이들 점들을 Bezier Curve로 연결하여 얻어진 얼굴형이다. 또, 우측의 그림은 이 얼굴형에 추출된 이목구비를 매핑시켜 얻어진 최종 결과물이다. 여기서 귀는 미리 만들어진 벡터 파일로서 매핑시 다른 이목구비와 함께 매핑시킨 것이다.

#### 4. 실험 결과 및 분석

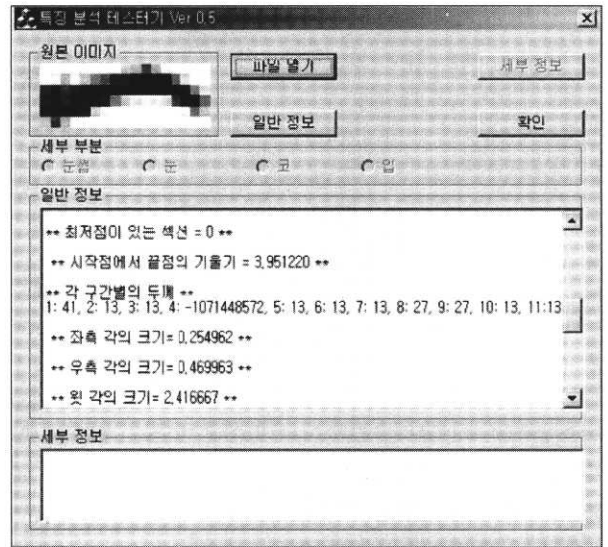
실험에 사용된 사진은 총 194장(남 : 97, 여 : 97)이었으며, 사진의 획득은 모두 인터넷을 통해 이루어졌다. 모두 입의의 유저로부터 직접 제출받은 것들이다. 따라서, 랜덤하게 추출되어졌다고 볼 수 있다. 단, 사진의 크기(가로, 세로 100픽셀 이상)와 정면 사진이라는 전제가 있었다. (그림 13)은 캐리커처 생성 시스템의 전체적인 외관이며 특징 추출이 이루어지고 있는 모습을 보여주고 있다. (그림 14)(a)는 입력된 이미지에서 추출된 눈썹의 특징분석은 눈썹을 한 예로 들어 정보를 추출하고 (그림 14)(b)는 유사도 계산에 의해 데이터베이스에서 추출된 이미지의 특징정보를 나타낸 것이다. 이와 같이 입력된 이미지에서 추출한 이목구비의 특징정보와 데이터베이스에서 유사도에 의해 추출한 이미지는 유사함을 알 수 있다.

실험 후 분석한 성공률은 다음과 같다.

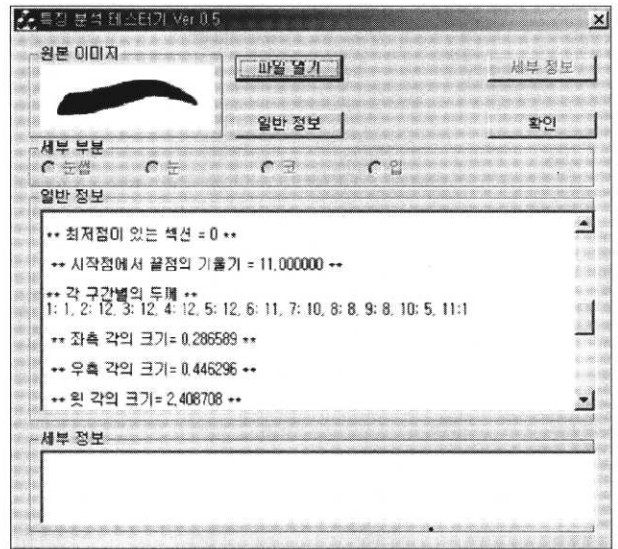
<표 6>은 얼굴영역 추출의 성공률이며, <표 7>은 각 이목구비 추출의 성공률을 도표화한 것이다. <표 7>에서 보는 바와 같이 눈의 위치 판별의 성공률은 비교적 높은 편(96%)이었으나 눈썹의 성공률은 그다지 높지 않은 것으로 나타났다. 그 이유는 눈썹은 머리카락에 가리워지거나 열은 눈썹을 가진 사람이 많고 또 조명에 따라서도 영향을 받기



(그림 13) 캐리커처 생성 시스템



(a)



(b)

(그림 14) (a) 입력된 이미지에서 추출한 눈썹의 특징분석  
(b) 유사도 계산에 의해 추출된 눈썹의 특징분석

때문이다. 이는 모양 특징정보 추출 시에도 영향을 미치는 것으로 나타났다 <표 8>. 코의 경우는 이목구비 중에서 가장 추출률이 낮은 것으로 나타났는데 여러모로 코의 특징 추출이 어려움을 보여주고 있다. 모양정보 추출에 있어서도 57%의 낮은 성공률을 보였다. 이는 코가 특히 조명변화에 민감하다는 것을 입증한다. 코의 위치는 눈과 입의 위치로 추정할 수 있으므로 <표 7>에서는 빠져있다. 모양 특징정보 추출에 있어서 눈, 입의 경우는 윤곽선이 뚜렷하기 때문에 비교적 높은 성공률(각각 88%, 95%)을 보인 반면 눈썹, 코, 턱의 경우는 모양 추출에 있어서 낮은 성공률을 나타내었다. 단, 턱의 성공률이 70%를 넘어섰다는 점과 캐리커처



생성에 있어서 핵심인 눈의 성공률이 비교적 높은 88%를 기록했다는 점에서 고무적이라 할 수 있겠다(<표 8>).

<표 6> 얼굴영역 추출 성공률

총 사진의 수	성 공	실 패
194	183 (94.3%)	11 (6.7%)

<표 7> 이목구비 추출 성공률(Localization)

이목구비	전체 사진개수	양쪽 모두 성공	한쪽만 성공	실 패
눈	183	161(88%)	15(8%)	7( 4%)
눈썹	176	79(45%)	41(23%)	56(32%)
입	176	143(81%)		33(19%)

<표 8> 모양 특징정보 추출 성공률

이목구비	전체 개수	VERY GOOD	GOOD	BAD
눈	176	92(52%)	63(36%)	21(12%)
눈썹	120	39(22%)	79(45%)	58(33%)
코	176	28(24%)	39(33%)	51(43%)
입	143	77(54%)	59(41%)	7( 5%)
턱	176	56(32%)	72(41%)	48(27%)

5. 결론 및 향후 연구

본 논문에서는 얼굴 사진을 이용하여 자신과 닮은 캐릭터의 얼굴을 자동으로 생성해 주는 시스템을 구현하였다. 이 시스템은 얼굴 사진으로부터 얼굴 영역만을 추출해 내는 모듈과 이목구비의 부분 영역을 추출하는 모듈과 각 이목구비별 특징정보를 추출하는 모듈 그리고 얻어진 특징정보를 이용해서 얼굴 모양(턱선)을 벡터화하고 최종적으로 얼굴형과 특징정보를 이용하여 DB로부터 검색되어진 이목구비를 특징정보에 맞게끔 적당한 위치에 배치하여 캐릭터의 얼굴을 생성하는 모듈을 포함하고 있다.

이 시스템을 통해 특정 인물의 사진을 이용하여 캐릭터를 자동으로 생성하는 테스트를 수행하여 만족할 만한 결과를 얻어내었다. 아직은 인식률의 부족과 이목구비 이미지의 부족으로 인해 정확성은 떨어지나 향후 발전될 것이라 기대된다.

향후 계획은 기울어진 얼굴과 측면 얼굴 그리고 흑인의 얼굴로도 캐릭터를 생성할 수 있도록 시스템을 개발해 나가는 것이다. 또한, 2D 캐릭터 얼굴에 3D 모션 데이터를 적용하여 표정을 부여할 수 있도록 개발해 나갈 계획이다.

참 고 문 헌

[1] Junji Nishino, Tomonori Kameyama, Haruhiko Shirai, To-

mohiro Odaka and Hisakazu Ogura, "Linguistic Knowledge Acquisition System on Facial Caricature Drawing System," IEEE International Fuzzy Systems Conference Proceedings, pp.22-25, August, 1999.

[2] Jun-ichiro Hayashi, Kazuhito Murakami, Hiroyasu Koshimizu, "Facial Caricaturing System PICASSO with Emotional Motion Deformation," Second International Conference on Knowledge-Based Intelligent Electronic System, pp.21-23, April, 1998.

[3] Francesco, G.B., De Natale, Daniele, D. Giusto and Fabrizio Maccioni, "A Symmetry-Based Approach To Facial Feature's Extraction," IEEE, 1998.

[4] G. HANAOKA, M. KANEKO, H. HARASHIMA., "Facial caricature by computer based on the style of individual human caricaturist," IEICE trans, Vol.J80-D-II, No.8, pp. 2110-2118, 1997.

[5] S. Iwashita and T. Onisawa, "A study on facial caricature drawing by fuzzy theory," Proceedings of FUZZ-IEEE '97, pp.933-938, 1997.

[6] J. Nishino, M. Shimasaki, H. Shirai, T. Odaka and H. Ogura, "facial caricature drawing system based on a fuzzy linguistic expression," Proceedings of 13rd. Fuzzy System Symposium, pp.465-468, 1997.

[7] Tominaga, M., Fukuoka, S., Murakami, K. and Koshimizu, H., "Facial Caricaturing with Mothion Cariaturing in PICASSO System," IEEE/ASME International Conference on Advanced Intelligent Mechatronics AIM'97, p.30, 1997.

[8] Jianming Hu, Hong Yan and Mustafa Sakalli, "Facial Feature Extraction from Head-Shoulder Images," The Fifth International Conference on Contol, Automation, Robotecs and Vision(ICARCV '98) Singapore, pp.9-11, December, 1998.

[9] Rein-Lien Hsu, A. K. Jain, M. Abdel-Mottaleb, "Face detection in color images," IEEE Trans on Pattern Analysis and Machine Intelligence, Vol.24, No.5, pp.696-706, 2002.

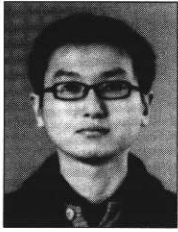
[10] Ming-Hsuan Yang, D. J. Kriegman, N. Ahuja, "Detecting faces in images : a survey," IEEE Trans on Pattern Analysis and Machine Intelligence, Vol.24, No.1, pp.34-58, 2002.

[11] 조용진, '얼굴, 한국인의 닳', 사계절 출판사, 1999.

[12] 김영일, 이용주, "얼굴피부색, 얼굴특징벡터 및 안면각 정보를 이용한 실시간 자동얼굴검출 및 인식시스템", 정보처리학회논문지B, 제9-B권 제4호, 2002.

[13] 이옥경, "얼굴 특징정보를 이용한 캐리커처 자동 생성 시스템", 숭실대학교 석사학위 논문, 2002.

[14] 최내원, 오해석, "실시간 칼라영상에서 객체추출 및 추적", 정보처리학회논문지B, 제10-B권 제1호, pp.81-86, 2003.



### 박연출

e-mail : parkpd@edunet.net

1997년 상지대학교 응용통계학과(이학사)

1999년 숭실대학교 컴퓨터학과(공학석사)

2001년 숭실대학교 컴퓨터학과(박사수료)

현재 숭의여자대학 인터넷정보과 겸임교수

관심분야 : 컴퓨터비전, 영상처리, 얼굴인식,

Facial Caricaturing



### 오해석

e-mail : oh@computing.soongsil.ac.kr

1975년 서울대학교 응용수학과(이학사)

1981년 서울대학교 계산통계학과(이학석사)

1989년 서울대학교 계산통계학과(이학박사)

1982년~현재 숭실대학교 컴퓨터학과 교수

관심분야 : Multimedia, Database, MIS